



## Cisco CloudCenter リリース 4.8.x トラブルシューティング ガイド

初回更新日: 2017 年 7 月 19 日

シスコシステムズ合同会社  
〒107-6227 東京都港区赤坂 9-7-1 ミッドタウン・タワー  
<http://www.cisco.com/jp>  
お問い合わせ先: シスココンタクトセンター  
0120-092-255 (フリーコール、携帯・PHS 含む)  
電話受付時間: 平日 10:00～12:00、13:00～17:00  
<http://www.cisco.com/jp/go/contactcenter/>

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザ側の責任になります。

対象製品のソフトウェアライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコとこれら各社は、商品性の保証、特定目的への準拠の保証と権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、シスコおよびその供給者は、このマニュアルの使用または使用できないことによって発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性がシスコまたはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワークボロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

Cisco および Cisco ロゴは、シスコや米国および他の国の関連会社の商標です。シスコの商標の一覧は <http://www.cisco.com/go/trademarks> [英語] を参照してください。Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2017 Cisco Systems, Inc. All rights reserved.

1. PostgreSQL のトラブルシューティング .....	2
2. CCO のトラブルシューティング .....	6
3. パラメータのトラブルシューティング .....	7
4. サポート チケットのチェックリスト .....	8

# PostgreSQL のトラブルシューティング

- OS のアップグレードとリポート後に PostgreSQL クラスタが起動しない
- PostgreSQL インスタンスが CCM との通信を停止している
- PostgreSQL セットアップのリカバリ
  - 前提条件
  - 一般的な HA のトラブルシューティングおよびリカバリの手順
  - マスターのホスト 1 へのスイッチバック
- PostgreSQL サービスがシャットダウンすると、データベースが同期しなくなり、サービスを再起動するとスロットル エラーになる

## OS のアップグレードとリポート後に PostgreSQL クラスタが起動しない

RHEL 7 上の MGMTPOSTGRES\_MASTER と MGMTPOSTGRES\_SLAVE の両方のサーバにパッチを適用し、両方のサーバをリポートした場合、両方のサーバがリポートしてからもクラスタは起動しないことがあります。これはクラスタが同期しておらず、リポート後に自動的に起動できないことが原因です。

システムのリポート後に、Pacemaker が起動可能であることを確認します。Pacemaker は、高可用性を確実にするために CloudCenter プラットフォームで使用されるクラスタ ユーティリティです。CloudCenter 4.8.0 からは、Pacemaker はリポートごとに開始するようになり、HA クラスタを手動で起動する必要がなくなりました。ただし代わりに、次の要件が正しく設定されていることを確認します。

- ファイアウォール ルールが正しく設定されていることを確認します。このユーティリティのファイアウォール ルールは、「[CCM Firewall Rules](#)」で説明されています。
- 各 PostgreSQL サーバに移動し、次のコマンドを入力して、データベースと HA 接続のステータスを確認します。

```
pcs status
```

- 両方のデータベース サーバの **PCSD ステータスがオンライン**であることを確認します。
- Corosync、Pacemaker、および PCSD サービスの **デーモン ステータスが アクティブ/無効**であることを確認します。
- PostgreSQL インスタンスが、次のセクション（「[PostgreSQL インスタンスが CCM との通信を停止している](#)」）で詳述されているとおりに CCM と通信していることを確認します。

## PostgreSQL インスタンスが CCM との通信を停止している

HA モードを使用しているときにこのようなシナリオ状況が発生した場合には、次の手順に従ってこの問題を訂正します。

1. Postgres サービスを開始します。

```
systemctl start postgresql-9.5
```

2. 必須ディレクトリ内に \*.lock ファイルが存在していないことを確認します。

```
cd /var/lib/pgsql/tmp/
# IF the PGSQ.L.lock is present, remove it using command
rm -rf *.lock
```

3. クラスタを停止し、それから起動します。

```
pcs cluster stop
pcs cluster start
```

4. クラスタのステータスを調べて、それが稼働していることを確認します。

```
pcs status
```

- それでも問題が解決しない場合は、dbmaster と PGMasterVIP を再起動します。

```
pcs resource restart dbmaster
pcs resource restart PGMasterVIP
```

## PostgreSQL セットアップのリカバリ



### PostgreSQL の要件

PostgreSQL のセットアップが後述の「前提条件」セクションに正確に従っていない場合、予期せぬ結果が生じる可能性があります。

## 前提条件

CloudCenter プラットフォームでは、次を使用する必要があります。

- PostgreSQL 9.5.6: CloudCenter プラットフォームは他のどのバージョンもサポートしていません。
- CentOS 7 または RHEL 7 OS (すべての PostgreSQL インスタンスが対象)。
- Corosync、Pacemaker、および pcsd サービスのためのアクティブなデーモン ステータス。



完全ロード RPM システムを使用するかまたは他の追加パッケージ (たとえば Fuse、EVM、Raid、または他の類似のパッケージ) をロードすると、予測しない結果になる場合があります。

## 一般的な HA のトラブルシューティングおよびリカバリの手順

場合によっては、設定と構成に応じて、PostgreSQL HA セットアップで、マスターとスレーブのインスタンスをネットワーク障害後にリブートまたは再接続する、あるいは現在のマスター データベース インスタンスがオフラインになったために HA フェールオーバーが発生するなどといったシナリオ状況になる可能性があります。このような場合には、クラスタ管理ソフトウェアがノード上でユーザの介入なしに PostgreSQL を開始できない可能性があります。

1. [クラスタのステータスを確認および是正するための一般的な管理タスク](#)
  - a. クラスタのステータスを調べて、それが稼働していることを確認します。

```
pcs status
```

- b. 使用例として (以下の pcs ステータス出力で示されるとおり)、エラーが原因でスレーブが開始しなかったという場合があります。たとえば、以下の出力に示すとおり、HA ソフトウェアがロック ファイルを生成し、変更があれば管理者がそれを確認してコミット/承認できるという既存のロック ファイル シナリオ状況により、スレーブが開始していないという場合があります。

```

Cluster name: cliqrdbcluster
Stack: corosync
Current DC: dbslave (version 1.1.15-11.e17-e174ec8) - partition
with quorum
Last updated: Thu Jun  1 10:16:56 2017
Last change: Thu Jun  1 09:42:23 2017 by hacluster via crmd on dbslave
2 nodes and 4 resources configured
Online: [ dbmaster dbslave ]
Full list of resources:
Resource Group: VIPGroup
    EC2SecondaryIP      (ocf::heartbeat:EC2SecIP):      Stopped
    PGMasterVIP         (ocf::heartbeat:IPaddr2):      Stopped
Master/Slave Set: mspostgresql [pgsql]
    Slaves: [ dbslave ]
    Stopped: [ dbmaster ]
Failed Actions:
* pgsql_start_0 on dbmaster 'unknown error' (1): call=15,
status=complete, exitreason='My data may be inconsistent. You have
to remove /var/lib/pgsql/tmp/PGSQL.lock file to force start.',
    last-rc-change='Thu Jun  1 09:42:24 2017', queued=0ms, exec=246ms
Daemon Status:
    corosync: active/disabled
    pacemaker: active/enabled
    pcsd: inactive/disabled

```

- c. 上記の手順 b で言及されているシナリオでは、停止しているノード上(上記の例では **dbmaster** という名前)にあるロック ファイルを削除(この例では `/var/lib/pgsql/tmp/PGSQL.lock` というロック ファイルを削除)、次のコマンドを実行します。

```
pcs resource cleanup mspostgresql
```

- d. **pcs status** を再度実行して、エラーがクリアされたことと、スレーブがオンラインに戻っていることを確認します。

これで PostgreSQL サービスは問題なく実行されるはずです。

## 2. HA インスタンスのレポート

管理者は、セキュリティ パッチやバージョンのアップグレードなどの特定のシステム管理アクションのために、HA インスタンスのレポートが必要になる場合があります。この場合、管理者はこのセクションに記載されている手順に従う必要があります。

- a. 各ノード上で **pcs status** を実行し、どのノードが**現行マスター**として作動しているかを判別します。

```

Cluster name: cliqrdbcluster
Stack: corosync
Current DC: dbmaster (version 1.1.15-11.e17-e174ec8) - partition with
quorum
Last updated: Fri Jun  2 03:23:22 2017
Last change: Thu Jun  1 10:42:41 2017 by root via crm_attribute on
dbslave
2 nodes and 4 resources configured
  Online: [ dbmaster dbslave ]
Full list of resources:
  Resource Group: VIPGroup
    EC2SecondaryIP      (ocf::heartbeat:EC2SecIP):      Started dbslave
    PGMasterVIP         (ocf::heartbeat:IPAddr2):        Started dbslave
  Master/Slave Set: mspostgresql [pgsql]
    Masters: [ dbslave ]
    Stopped: [ dbmaster ]
  Failed Actions:
    * pgsql_start_0 on dbmaster 'unknown error' (1): call=15,
status=complete,
  exitreason='My data may be inconsistent. You have to remove
/var/lib/pgsql/tmp/PGSQL.lock file to force start.',
  last-rc-change='Thu Jun  1 10:43:15 2017', queued=0ms,exec=263ms
  Daemon Status:
    corosync: active/disabled
    pacemaker: active/enabled
    pcsd: inactive/disable

```

上記の出力は、**dbslave** というノードが現行マスターであることを示しています。

- b. 現行のスレーブ ノードをリポートします。上記の例では、**dbmaster** というノードをリポートします。
- c. リポート後に、上記の「クラスタのステータスを確認および是正するための一般的な管理タスク」セクションに示されている手順に従って、ノードにエラーがあるかどうかを確認します。クリーンアップが必要な場合には、このセクションに記載されている手順に従います。
- d. 現行マスターであるノードをリポートします。この例では、**dbslave** というノードが現行マスターです。
- e. リポートが実行されたら、現行スレーブであるノード上で **pcs status** コマンドを実行します。この例では **dbmaster** というノードが現行スレーブです。HA フェールオーバーが行われ、ノードが新しいマスターになったことがわかります。
- f. 古いマスターを再度リポートしたら、上記の「クラスタのステータスを確認および是正するための一般的な管理タスク」セクションに示されている手順に従って、エラーを確認して、エラーがあればクリーンアップし、リポートしたノードがスレーブとしてクラスタに再度参加できるようにします。

## マスターのホスト 1 へのスイッチバック

ホスト 1 (マスター) とホスト 2 (スレーブ) があり、フェールオーバーが行われてホスト 2 がマスターになるというシナリオを想定します。

その後、マスターをホスト 1 にスイッチバックしたいという状況になったとします。このセクションでは、このタイプのスイッチオーバーのプロセスについて説明します。

1. いずれかの PostgreSQL インスタンスから、次のコマンドを発行します。

```
pcs cluster stop host2
```

この時点で、ホスト 1 が再びマスターになります。

2. PCS ステータスを調べて、発生する可能性があるエラーがないかを確認します。

```
pcs status
```

3. 次の失敗したアクション エラーがある場合には、以下のようになります。

```
Failed Actions:
* pgsql_start_0 on dbmaster 'unknown error' (1): call=15, status=complete,
exitreason='My data may be inconsistent. You have to remove
/var/lib/pgsql/tmp/PGSQL.lock file to force start.',
last-rc-change='Tue Jun 13 14:59:43 2017', queued=0ms, exec=224ms
```

次のコマンドを実行して、HA を機能状態に戻します。

```
pcs resource cleanup mspostgresql
```

4. クラスタのステータスを再度調べて、エラーが示されておらず、すべてのサービスが稼働状態であることを確認します。

```
pcs status
```

この時点で、示されているエラーはないことになります。

## PostgreSQL サービスがシャットダウンすると、データベースが同期しなくなり、サービスを再起動するとスロットル エラーになる

現行ノード上で PostgreSQL および PCS クラスタを停止して再起動するシーケンスについては、「[HA Mode Backup and Recovery](#)」に詳しく説明されています。HA モードでのデータベースの停止または起動については、そのページに記載されている手順に従ってください。

HA サービスで問題がある場合には、追加のコンテキストについて、「[Graceful PostgreSQL Reboot](#)」を参照してください。

# CCO のトラブルシューティング

## AMPQ と CCO の設定後に両方のサーバ間での通信が確立されない

すべてのポート関連のニュアンスや要件は、「[Install](#)」セクションの「[Phase 2: Configure Firewall Rules](#)」を参照してください。

CCO サーバと AMQP サーバの間の最も一般的な通信の問題は、RabbitMQ 構成が適切なタイミングで完了していないということです。RabbitMQ 構成は、次のいずれかの実行前に完了する必要があります。

- 仮想アプライアンスのインストール: AMQP アプライアンス イメージのリポート
- 手動インストール: AMQP サーバ上でのコア インストーラのリポート

次のコマンドを実行してここにリストされている出力を確認することで、RabbitMQ 構成が正常に完了したことを確認できます。



```
# rabbitmqctl list_vhosts
Listing vhosts ...
/
/cliqr

# rabbitmqctl list_users
Listing users ...
cliqr [administrator]
cliqr_worker []

# rabbitmqctl list_permissions -p /cliqr
Listing permissions in vhost "/cliqr" ...
cliqr .*      .*      .*
cliqr_worker .*    .*      .*
```

## パラメータのトラブルシューティング

### 概要

CloudCenter プラットフォームでは、独自のパラメータを定義するか(「Using Parameters」を参照)、または CloudCenter 定義のパラメータを使用することができます。

開発またはテスト デバッグの目的のために、CloudCenter プラットフォームはトラブルシューティング パラメータを提供しており、ユーザは必要に応じて必要なパラメータを呼び出すことができます。

### アプリケーションの障害を無視するパラメータ

**cliqrignoreAppFailure** パラメータは、障害時にノードを終了せず、すべての失敗したアプリケーション導入ノードの実行を継続することができます。現在はセキュリティグループとともにアプリケーション プロファイルが削除されます。



#### 注意

導入の障害のみをトラブルシューティングするには、**cliqrignoreAppFailure** グローバル パラメータを使用します。このパラメータは、初期のデバッグ専用であり、アプリケーションが正しく機能したらセキュリティ上の理由から削除する必要があります。

すべての障害があったノードをあらゆるアプリケーションに対して実行するように設定するには、次の手順に従います。

1. アプリケーションをモデリングします。
2. アプリケーションを編集します(「Application Tasks」 > 「Edit/Update」を参照してください。)
3. **cliqrignoreAppFailure** グローバル パラメータを設定します
  - a. 「Topology Modeler」 > 「Global Parameters」にアクセスします。
  - b. [パラメータの追加(add a parameter)] のリンクをクリックして、[パラメータ(Parameter)] ドロップダウン セクションを開きます。

## Model a N-Tier Execution Application Profile

Basic Information | Global Parameters | Topology Modeler

If you would like to add additional parameters, this is the place.

Global Parameters | add a parameter »

▼ Parameter

Parameter Name: cliqrlgnoreAppFailure 19

Display Name: Ignore App Failure 22

Help Text: The app will not terminate nodes during a failure 206

Type: string Max Length: 255

Default Value: true 251

User Options:

- Should this parameter be visible to the user?
- Should this parameter be editable by the user?
- Should this parameter be optional?

Save as Template | Save as App | Cancel

- c. 各フィールドに次の値を追加します。
- i. パラメータ名: **cliqrlgnoreAppFailure**
  - ii. 表示名: **Ignore App Failure**
  - iii. ヘルプ テキスト: **アプリケーションは障害時にノードを終端しません**
  - iv. タイプ: **文字列**
  - v. デフォルト値: **true**
  - vi. 導入システムに応じて、「User Options」をチェックします(追加情報については、「Using Parameters」 > 「Granular Control for User-Defined Parameters」を参照してください)。
4. 導入システムに応じてプロファイルをアプリケーションまたはテンプレートとして保存します。
5. **アプリケーションを導入します**。**cliqrlgnoreAppFailure** グローバル パラメータの設定は、[グローバル パラメータ(Global Parameters)] セクションの [Ignore App Failure] に [true] として表示されます。

### Global Parameters

Specify values for parameters that the application needs in order to run.

Ignore App Failure:

true

The app will not terminate nodes during a failure

これでこのアプリケーションの **cliqrlgnoreAppFailure** グローバル パラメータは設定され、障害があったどのノードもすべて終了しません。

# サポート チケットのチェックリスト

- チェックリスト
- 高度なトラブルシューティング
- サポート チケットのファイル化

## チェックリスト

オンボード、移行、管理、制御のいずれかの段階で導入システムが無効または無応答になった場合は、サポート チケットをファイル化してください。

状況を正しく診断するために、次の情報をサポート チケットに添付または記入してください。

番号	項目
1	<p>問題に発展した経緯を詳細に文書化します。次に例を示します。</p> <ul style="list-style-type: none"> <li>• 何を変更、設定、追加、または削除したか。</li> <li>• 導入システムの設定を変更した可能性がある、整合性がないまたは非互換のクラウド セキュリティまたはモニタリング ツールをインストールしたか。</li> <li>• イメージ マッピングを変更したか。そうであるなら、手順のメモを作成したか。</li> <li>• 新しい CCO を追加したか。そうであるなら、手順のメモを作成したか。</li> <li>• CCO の設定を変更したか。そうであるなら、手順のメモを作成したか。</li> <li>• 新しいイメージ マップに最新バージョンの CloudCenter イメージがインストールされているか。 CloudCenter プラットフォームと同じバージョンの CloudCenter イメージがあるか。</li> </ul>
2	<p>ダッシュボード上の [クラウドのヘルス &amp; ステータス (Cloud Health &amp; Status)] のサークルが赤に変化している場合は、対象のクラウドが作動して機能していることを確認します。 対象のクラウドがダウンしている場合には、問題を診断し、サークル ライトが緑色になったときに再び作動することを確認します。</p>
3	<p>対象のクラウドが作動している場合、Tomcat サービス実装に問題がある可能性があります。 Tomcat を再起動して、問題の考えられる原因があるかどうかを判別します。</p>
4	<p>導入システムのすべてのログをバックアップします。 詳細な診断のために、関連するログとスナップショットを CloudCenter サポートに送信します。</p>

## 高度なトラブルシューティング

番号	項目
1	<p>問題があるサーバで <code>jstack</code> コマンドを実行し、ファイルにログを保存します。保存したファイルをサポート チケットに添付します。</p> <pre><code>jstack TomcatServiceProcessID &gt; LogFileName</code></pre>
2	<p>問題があるサーバで <code>ps-efL</code> コマンドを実行し、ファイルにログを保存します。ファイルをサポート チケットに添付します。</p> <pre><code>ps-efL TomcatServiceProcessID &gt; LogFileName</code></pre>

## サポート チケットのファイル化

サポート チケットに記入して、CloudCenter サポート チームに問い合わせてください。